REMARKS

Claim 1-35 are pending. Claims 1, 12, 22 and 32 are independent.

Applicant amended independent claims 12, 20, 30 and 32 to correct several typographical errors. Applicant also amended claims 8, 19, 29 and 31 to correct several antecedent problems.

In addition, applicant added new claim 36, which depends from independent claim 1, reciting that the interface of the second processor is configured to send the task to the second processor when the second processor is in slave mode, and to send a result of the task or another task to the first processor when the second processor is in master mode. Support for this added feature is provided, for example, at page 4, line 22, to page 5, line 20, of the originally filed application. Applicant also added new claims 37-39, each depending from independent claims 12, 22 and 32, respectively, reciting features similar to those recited in new claim 36.

The examiner stated that the title of the application is not descriptive. Applicant amended the title to "Interconnected processors configured for co-processing functionality."

The examiner rejected claims 29-31 under 35 U.S.C. §112, second paragraph on the ground that there is an insufficient antecedent basis for the "apparatus" recited in claim 29, and the "method" recited in claims 30-31.

Applicant amended claims 29-31 to recite in the preamble of those claims the term "article". As the amended claims depend from claim 22 which recites "article", applicant submits that claims 29-31 now have a sufficient antecedent basis for the amended term "article."

The examiner rejected claims 22-31 under 35 U.S.C. §101 on the ground that the subject matter of the claims is directed to non-statutory subject matter. The examiner stated:

> Last paragraph of page 12 of the specification provides intrinsic evidence that applicant's clear intent is that the broadest reasonable interpretation of the claim term "computer readable media" is that the term is to encompass propagated signals. Propagated signals are a form of energy. Energy is not one of the four categories of the invention and therefore the claims are not statutory. Energy is not a series of steps or acts and thus is not a process. Energy is not a physical article or object and as such is not a machine or manufacture. Energy is not a combination of substances and therefore is not a composition of matter. Accordingly, the rejection of claims 22-31 are non-statutory. (Office Action, paragraph 9, page 3)

While applicant believe that the term "[a]n article" preceding the language "computer readable medium" in the preamble of claim 22 clearly indicates that applicant's claim is not directed to propagated signals, for the sake of expediting prosecution, applicant amended claim 22 to recite "[a]n article comprising a machine-readable storage device" which, as indicated at page 12 of applicant's Specification, does not include propagated signals. Accordingly, the rejection under 35 U.S.C. §101 has been overcome and should be removed.

The examiner rejected claims 1-3, 12-14, 22-24 under 35 U.S.C. §102(b) as being anticipated by U.S. Patent No. 5,068,821 to Sexton. Additionally, the examiner rejected claims 4-10, 15-21 and 25-31 under 35 U.S.C. §103(a) as being unpatentable over Sexton in view of the reference "The Next Generation of Intel IXP Network Processors" by Adiletta et al. Further, the examiner rejected claim 11 under 35 U.S.C. §103(a) as being unpatentable over Sexton in view of U.S. Patent Publication No. 2002/0143998 to Rajagopal et al., rejected claim 32 under 35 U.S.C. §103(a) as being unpatentable over Sexton in view of U.S. Patent Publication No. 2002/0009079 to Jungck et al, and rejected claims 33-35 under 35 U.S.C. §103(a) as being unpatentable over Sexton in view of Jungck and further in view of Adiletta.

Specifically, with respect to independent claim 1, the examiner stated:

> 12. As per claim 1, Sexton teaches a method of co-processing, comprising:
>
> connecting an interface of a first processor (element 110) to an interface of a second processor (element 120) using a bus, the interface of the second processor being configurable to place the second processor in a slave processing mode or a master processing mode (see e.g. Fig. 2. Also, see e.g. col. 5, lines 47-67 to col. 6, lines 1-64); and
>
> sending a task from the first processor to the second processor through the bus, the task comprises an instruction that places the second procesor in a slave processing mode (see e.g. col. 5, lines 47-67, wherein the start command is the instruction). (Office Action, paragraph 12, page 4)

Applicant respectfully disagrees with the examiner's contentions.

Applicant's independent claim 1 recites "[a] method of co-processing, comprising: connecting an interface of a first processor to an interface of a second processor using a bus, the

interface of the second processor being configurable to place the second processor in a slave processing mode or a master processing mode."

Sexton, does not describe these features. Rather, Sexton describes programmable logic controller that includes a function block processor 110 for processing function block instructions and a bit processor for processing Boolean instructions (FIG. 2, Abstract, and col. 2, lines 36-59). Sexton explains:

> **Bit processor 120 is operable in two modes, namely a master mode and a slave mode. A master bit is written to master/slave mode register 121 to place bit processor 120 in the master mode. Alternatively, to place bit processor 120 in the slave mode, a slave bit is written to master/slave mode register 121 of the bit processor. When bit processor 120 is placed in the slave mode in this manner, function block processor 110 has control of user program RAM 135. When function block processor 110 desires to use bit processor 120, function block processor 110 writes a start command. Bit processor 120 then takes control of user RAM 135 by writing a master bit into the master/slave register. Concurrently, tristate buffers (not shown) are engaged to shut off function block processor 110 from the data and address busses connecting processor 110 to the remainder of programmable logic controller 100. Function block processor 110 is permitted to service interrupts and execute direct memory access (DMA) cycles in this mode. However, processor 110 is otherwise held in an inactive state by bit processor 120 asserting a WAIT signal on the WAIT signal line coupling bit processor 120 to function block processor 110 as shown in FIG. 2. In this master mode, function block processor 110 has no access to user program RAM 135. (emphasis added, col. 5, line 43, to col. 6, line 4)**

And:

> **The operation of bit processor 120 in master mode is now discussed in more detail through reference to the operational flow chart of the master mode shown in FIG. 3. To initiate the master mode, function block processor 110 writes a start command to bit processor 120 as per block 200. A master bit is then written to master/slave mode register 121 as per block 205. Once this starting operation is performed, bit processor 120 isolates itself from function block processor 110 and asserts a WAIT signal on the WAIT line as per block 210. In response to the WAIT signal, function block processor 110 then executes an instruction which causes the function block processor to wait until bit processor 120 removes or de-asserts the WAIT signal. That is, function block processor 110 is permitted to respond to interrupts, but is forced to return to waiting once servicing of the interrupts is complete.**
>
> **Bit processor 120 decodes a command in user program RAM 135 as per block 215. If at decision block 220 a determination is made that the current command is not a function block command with zero power flow or is not a function block command of the type having an OPCODE which must be processed by function block processor, then bit processor 120 executes the**

> **current command as per block 222. At block 224, bit processor 120 then advances the program counter 122 to point to the next command. Flow then continues back to block 200. However, if at descision block 220 a determination is made that the current command is a function block command with zero power flow or a function block command of the type having an OPCODE which must be processed by function block processor, then flow continues to block 225 at which a slave bit is written to mode register 121. This action returns bit processor 120 back to the slave mode. Bit processor 110 also de-asserts the WAIT signal to return control to function block processor 110 as per block 230. The function block processor 110 then reads the instruction pointer from bit processor 120 and thus is provided a reference to its OPCODE in user program RAM 135 as per block 240. It is noted that the OPCODE of the current command has no meaning to function block processor 110. However, the information following the OPCODE does have meaning to function block processor 110. Bit processor 120 then increments or otherwise adjusts program counter 122 to point to the address in user program RAM 135 of this next OPCODE as per block 235 thus updating an instruction pointer. (col. 6, lines 15-63)**

Thus, it is the bit processor 120 that places itself in master mode (i.e., when the bit processor has control of the resources of the controller 100) or in slave mode (i.e., when the function block processor 110 has control of the resources, such as RAM 135, of Sexton's controller 100). For example, bit processor 120 places itself in master mode in response to a start command written by function block processor 110 (col. 5, lines 56-60). But at no point does Sexton describe that an interface of the bit processor 120 or of function block 110 is configured to place the bit processor 120 in master or slave mode. Indeed, Sexton does not even disclose interfaces used in conjunction with either of the processors.

Accordingly, Sexton fails to disclose or suggest at least the feature of "connecting an interface of a first processor to an interface of a second processor using a bus, the interface of the second processor being configurable to place the second processor in a slave processing mode or a master processing mode," as required by applicant's independent claim 1. Applicant's independent claim 1 is therefore patentable over the cited art.

Claims 2-11 depend from independent claim 1, and are therefore patentable for at least the same reasons as independent claim 1.

Applicant's independent claims 12 and 22 recite "a first processor having an interface connected to an interface of a second processor using a bus, the interface of the first processor

being configurable to place the first processor in a slave processing mode or a master processing mode," or similar language. For reasons similar to those provided with respect to independent claim 1, at least this feature is not disclosed by the cited art. Accordingly, applicant's independent claims 12 and 22 are patentable over the cited art.

Claims 13-21 depend from independent claim 12, and are therefore patentable for at least the same reasons as independent claim 12. Claims 23-31 depend from independent claim 22, and are therefore patentable for at least the same reasons as independent claim 22.

As noted, the examiner rejected independent claim 32 as being unpatentable over Sexton in view of Jungck.

Applicant's independent claim 32 recites "[a] network router, comprising: a network co-processing system, the network co-processing system comprising: a first processor having an interface; and a second processor having an interface connected to the interface of the first processor by a bus, the interface of the second processor being configurable to place the second processor in a slave processing mode or a master processing mode, an input line connecting the network co-processing system to a first network; and an output line connecting the network co-processing system to a second network."

For reasons similar to those provided with respect to independent claim 1, Sexton does not disclose or suggest at least the feature of "a second processor having an interface connected to the interface of the first processor by a bus, the interface of the second processor being configurable to place the second processor in a slave processing mode or a master processing mode, and the Jungck reference is not seen to cure these deficiencies in Sexton.

Because neither Sexton nor Jungck discloses or suggests, alone or in combination at least the feature of "a second processor having an interface connected to the interface of the first processor by a bus, the interface of the second processor being configurable to place the second processor in a slave processing mode or a master processing mode," applicant's independent claim 32 is therefore patentable over the cited art.

Claims 33-35 depend from independent claim 32, and are therefore patentable for at least the same reasons as independent claim 32.

It is believed that all the rejections and/or objections raised by the examiner have been addressed.

In view of the foregoing, applicant respectfully submits that the application is in condition for allowance and such action is respectfully requested at the examiner's earliest convenience.

All of the dependent claims are patentable for at least the reasons for which the claims on which they depend are patentable.

Canceled claims, if any, have been canceled without prejudice or disclaimer.

Any circumstance in which the applicant has (a) addressed certain comments of the examiner does not mean that the applicant concedes other comments of the examiner, (b) made arguments for the patentability of some claims does not mean that there are not other good reasons for patentability of those claims and other claims, or (c) amended or canceled a claim does not mean that the applicant concedes any of the examiner's positions with respect to that claim or other claims.

The fees in the amount of $200, for excess claims, are being paid concurrently on the Electronic Filing System (EFS) by way of Deposit Account authorization. Please apply any other required fees to deposit account 06-1050, referencing the attorney docket number shown above.

Respectfully submitted,

Date: Feb. 1, 2007

Ido Rabinovitch
Attorney for Intel Corporation
Reg. No. L0080

PTO Customer No. 20985
Fish & Richardson P.C.
Telephone: (617) 542-5070
Facsimile: (617) 542-8906